MERTZ: A Quest for a Robust and Scalable Active Vision Humanoid Head Robot

Lijin Aryananda and Jeff Weber
MIT Computer Science and Artificial Intelligence Laboratory
Cambridge, Massachusetts, 02139, USA
{*lijin, jaweber*}*@csail.mit.edu*

We present the design and construction of MERTZ, an active-vision humanoid head robot, with the immediate goal of having the robot runs continuously for many hours a day without supervision at various locations. We address how the lack of robustness and reliability lead to limitations and scalability issues in research robotic platforms. We propose to attend to these issues in parallel with the course of robot development. Drawing from lessons learned from our previous robots, we incorporated various fault prevention strategies into the electromechanical design. We have implemented a preliminary system, integrating sensorimotor, vision, and audio in order to test the full range of all degrees of freedom and enable the robot to engage in simple visual and verbal interaction with people. We conducted a series of experiment where the robot ran for 82 hours within 9 days at different public spaces. The robot interacted with a large number of passersby and collected at least 100,000 face images of at least 600 individuals within 4 days. We learned various lessons involving the robustness of current design and identified a set of failure modes. Lastly, we present the long term research direction for the robot.

Keywords: humanoid robot, social interaction, reliability

1. Introduction

Despite tremendous research progress in humanoid robotics over the past decade, it still remains challenging to develop a robot capable of performing in a robust and reliable manner. Research humanoid robots generally have a limited average running period and are mostly demonstrated through video clips, which provide a relatively narrow perspective on the robot's capabilities and limitations. Moreover, given the substantial challenges involved, it is common to employ shortcuts so as to allow for initial progress. Temporary solutions, such as covering up a wooden door that is often mis-detected as skin segment or setting up dark curtains to minimize lighting variations, are quite typical. Such simplifications, while far from reducing the environment to a blocks world, naturally raise some concerns. For instance, will the results from a given setting scale to a more complex environment? Could these shortcuts be hampering potential progress?

As accurately described by Bischoff and Graefe [2], robot reliability has not received adequate attention in most research projects on humanoid robotics. One

possible cause could be a false belief that when the time comes for robots to be expedited, someone else will eventually address the issue of reliability. In this paper, we propose to address the issue of robustness and reliability in parallel with the course of robot development for the following reasons. First, having a more reliable robotic platform suggests important research directions, and also provides a technical foundation on which to study them. As a concrete illustration, a robot that can run for days at a time under various conditions would be able to accommodate long term online learning processes and experiments. Moreover, a robot capable of continuous operation suggests a variety of interesting research directions in social interaction, especially vis a vis longer-term dynamics. Supposing that our pet dogs were awake for only several hours per week, imagine how impoverished our social interactions would be. Second, addressing the issue of reliability is likely to motivate further work on scalable solutions. Consider, for example, the large amount of streaming sensor data that would be collected over several days of experiments; such large volumes of data would require novel software solutions and learning algorithms. Moreover, algorithms are often tuned to the particular setting of the robot laboratory. Obliging the robot to perform at different locations would encourage innovative approaches that are scalable to other settings and environments.

In this paper, we present the design and development of MERTZ, an active-vision head robot platform for exploring scalable robot learning in a social context. The long term goal is to have the robot be situated in *our* social settings, operating continuously for long periods of time in different public spaces, interacting with various people. Mertz will have differing degrees of familiarity with human interlocutors. Much like human infants, we intend for Mertz to be able to detect frequently occurring perceptual events and associate co-occuring external and self-initiated experience, through interacting with human mentors. The notion that the human developmental process should play a significant role in the attempt to emulate human intelligence and the associated idea of a child machine learning from human teachers both date back at least to Alan Turing's 1950 paper "Computing Machinery and Intelligence" [0]. From the point of view of Human Computer Interaction (HCI), a robot that is able to *remember* who you are and certain words that you have taught to the robot after spending some time with you is more likely to generate a more dynamic interaction. Moreover, we believe that the robust ability to filter through a vast amount of sensory input received from the complex world, consistently detect and *remember* relevant associations would be a useful framework to implement in the ongoing process of developing humanoid robot intelligence.

To this end, we report on the design and development steps directed toward our first milestone, i.e. having the robot operate continuously for many hours in different public spaces and engage in simple visual and verbal interaction with a large number of people with minimal constraints. Drawing from lessons learned from our previous robots, we incorporated various fault prevention strategies into the electromechanical design. We have implemented a preliminary system, integrating sensorimotor, vision, and audio to allow simple interaction with humans, where the

robot orients to and tracks salient visual targets, such as human faces and bright-colored toys, and tries to mimic phoneme sequences extracted from robot-directed speech. We conducted a series of experiments where the robot ran for 82 hours within 9 days at different locations each time, in order to test the robustness of current design and identify failure modes. The robot was placed at various public locations with a written sign requesting people to interact with it for the experiment. While the robot interacted with a large number of passersby, visual and audio data were collected under varying conditions at different times of day and locations for future learning experiments.

In the following section, we illustrate a set of design challenges and criteria, as well as the scope of the project. Section 3 describes the robotic platform. Section 4 discusses the mechanical design of the robot, followed by details of the computational hardware and software for the preliminary integrated system in section 5. We report experimental results and observed failure modes in section 6. We sketch out the long term research direction of the project in section 7. Section 8 provides comparisons with related work, followed by conclusion in section 9.

2. Design Challenges and Research Scope

As human is an incredibly complicated creature, building humanoid robots is a struggle of dealing with a high level of complexity with limited resources and a large set of constraints. Failures may occur at any point in the intricate dependency and interaction among the mechanical, electrical, and software systems. Each degree of freedom of the robot may fail because of inaccurate position/torque feedback, loose cables, obstruction in the joint's path, processor failures, stalled motors, error in initial calibration, and various other sources. Even if all predictable problems are taken into account during design time, emergent failures often arise due to unexpected features in the environment. Perceptual sensors particularly suffer from this problem. The environment is a rich source of information for the robot to learn from, but is also plagued with a vast amount of noise and uncertainty. Naturally, the more general the robot's operating condition needs to be, the more challenging it is for the robot to perform its task.

It is important to note that in this project, we will not be investigating fundamental engineering solutions for fault tolerance in robotics in general. Instead, we aim to conduct artificial intelligence research with a humanoid robot platform with the awareness for the importance of robustness and reliability. We believe that raising the bar on the benchmark for robustness and generality of operating condition will generate useful insights. We have established a set of criteria and invested efforts to satisfy them during the design and construction process. First, the robot must be able to run continuously without supervision for many hours at a time. Thus, robot must be immune against typical incidents, such as power shutdown, power cycle, human error in start-up sequences, etc. Modularity in subsystems and maximizing autonomy at each control level are crucial in order to minimize chaining

of failures, leading to catastrophic accidents. The mechanical, electrical, and software components must monitor each other to identify problems and allow the robot to shut itself down whenever necessary. All software programs must be developed to run for many hours and thus free of occasional bugs and memory leaks. In addition to fault related issues, the robot must be easily transported and set up at different locations. The start-up procedure must be streamlined such that the robot can be turned on and off with minimum effort. In past projects, such a trivial issue had generated enough hesitation to turn on the robot frequently. Lastly, we believe that long exhaustive testing process in different environmental conditions is necessary to explore the full range of possible failure modes. We plan to continue identifying failure sources throughout the rest of the robot development process in order to implement the appropriate fault detection and increase fault tolerance.

3. Robotic Platform

Mertz is an active-vision head robot with thirteen degrees of freedom (DOF), using nine brushed DC motors for the head and four RC servo motors for the face elements (see Figure 1). The head pans and uses a cable-drive differential to tilt and roll. The eyes pan individually, but tilt together. The neck also tilts and rolls using two Series Elastic Actuators [20] configured to form a differential mechanism. The eyelids are coupled as one axis. Each eyebrow and lip is independently actuated. As is, Mertz is limited to using gaze direction and vocalization to actuate in the world and demonstrate its competence. In the near future, an arm-like component may be added in order to expand the repertoire of possible learning tasks, such as poking or pointing to objects.

The robot perceives visual input using a Point Grey OEM Dragonfly digital camera per eye. The robot receives proprioceptive feedback from both potentiometer and encoder mounted on each axis. The robot also takes in audio input using GN Netcom VA-2000 voice array desk microphone, allowing multiple people to speak to the robot. The robot's vocalization is produced by the DECtalk phoneme-based speech synthesizer using regular PC speakers.

A number of primary DOFs are dedicated to emulate each category of human eye movements, i.e. saccades, smooth pursuit, vergence, vesticular-ocular reflex, and opto-kinetic response [5]. The expressive element of Mertz's design is essential for the long term project goal, which involves having the robot learn from people through social interaction interface. As a tradeoff between complexity and robustness, we attempted to minimize the total number of DOFs while maintaining sufficient expressivity. The robot is mounted on a portable wheeled platform that is easily moved around and can be turned on anywhere by simply plugging into a power outlet.

4. Mechanical Design

Mertz was mechanically designed with the goal of having the robot be able to run for days at a time without supervision. Drawing from lessons from previous robots,

Fig. 1. Multiple views of MERTZ, an active-vision humanoid head robot with 13 DOF. The head and neck have 9 DOF. The face has actuated eyebrows and lips for generating facial expressions. The robot perceives visual input using two digital cameras and receives audio input using a desk voice-array microphone, placed approximately 7 inches in front of the robot. The robot is mounted on a portable wheeled platform that is easily moved around and can be turned on anywhere by simply plugging into a power outlet.

we incorporated various failure prevention and maintenance strategies, as described below.

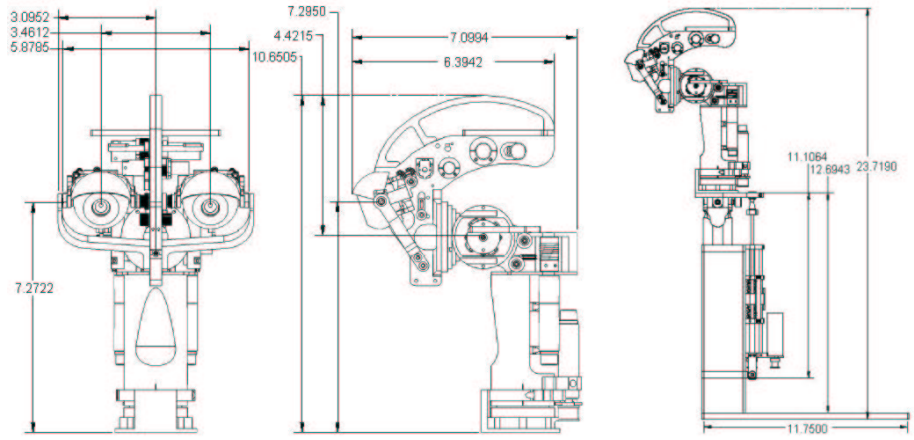

Fig. 2. The overall head dimension is 10.65 x 6.2 x 7.1 inches and weighs 4.25 lbs. The Series Elastic Actuators extend 11.1 inches below the neck and each one weighs 1 lb.

4.1. *Compact design to minimize total size, weight, and power*

A high-priority constraint was placed during the early phase of the design process to minimize the robot's size and weight. A smaller and lighter robot requires less torque from the motors to reach the same velocity. Also, the robot is less prone to overloading causing overheating and premature wear of the motors. The overall head size is 10.65 x 6.2 x 7.1 inches (see Figure 2) and weighs 4.25 lbs. which is supported by a box frame. The Series Elastic Actuators, which bear the weight of the head at the lower neck universal joint, extend below it 11.1 inches. MERTZ's compact design is kept light by incorporating nominal light alloy parts, which retaining stiffness and durability for their small size. Titanium, as an alternative to aluminum, was also used for some parts in order to minimize weight without sacrificing strength.

A cable-drive differential was implemented to provide the head roll and tilt in a small and compact design, as one axis provides two degrees of freedom. In contrast to a traditional gear train differential, a cable differential has increased efficiency, has significantly less mass, and eliminates backlash. Additionally, this design shares the load between two motors which reduces the amount of strain on each motor and either doubles the power of the joint as both motors work in unison, or allows for smaller motors to be used.

4.2. *Force sensing for compliancy*

Two linear Series Elastic Actuators (SEA) [20] are used for the differential neck joint, a crucial axis responsible for supporting the entire weight of the head. As shown in Figure 3, each SEA is equipped with a linear spring that is placed in series with the

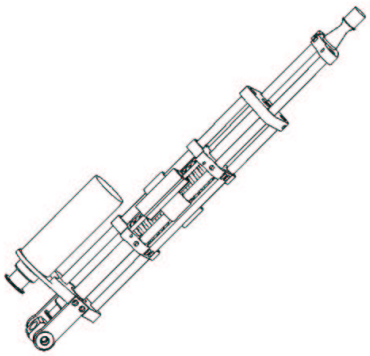

Fig. 3. The Series Elastic Actuator is equipped with a linear spring that is placed in series between the motor and load.

motor, which act like a low-pass filter reducing the effects of high impact shocks, while protecting the actuator and the robot. These springs also, in isolating the drive nut on the ball screw, provide a simple method for sensing force output, which is proportional to the deflection of the spring (Hooke's Law, F = kx where k is the spring constant). This deflection is measured by a linear potentiometer between the frame of the actuator and the nut on the ball screw. Consequently, force control can be implemented which allows the joint to be compliant and to safely interact with external forces in the environment.

The ball screw that is used in this linear actuator, which is 8mm in diameter with a pitch of 1mm, is equivalent to using a 79 to 1 planetary gear head. Although the linear actuator is longer, it provides necessary advantages in this case. Ball screws are over 90% efficient compared to roughly 80 % for a high quality planetary gear head. They are also back drivable without the risks associated with back driving a planetary or spur gear head, yet they require only a small amount of power to hold static postures. Additionally the SEA's will maintain position of the head when motor power is turned off. Collapsing joints upon power shutdown is a vulnerable point for robots, especially large and heavy ones.

Another concern for allowing the robot to run for long periods of time is heat generated by the motors. With running the robot for periods of 12 hours or more, only the linear actuator motors are generating a significant amount of heat. To deal with this we have been working on software issues to reduce noise in the system, which creates cogging in the motors generating heat, and designed heat sinks to place directly on the motors.

4.3. *Safeguarding position-controlled axes*

The rest of the DOFs rely on position feedback for motion control and thus are entirely dependent on accurate position sensors. Incorrect reading or faulty sensors could lead to a serious damage to the robot, so redundant relative encoder and potentiometer are utilized in each joint. The potentiometer provides absolute position measurement and eliminates the need for calibration routines during startup. Both sensors serve as a comparison point to detect failures in the other. Each joint is also designed to be back drivable and equipped with a physical stop in order to reduce failure impacts.

4.4. *Electrical cables and connectors*

Placement of electrical cables is frequently an afterthought in robot designs, as broken or loose cable is one of the most common failure sources. Routing over thirty cables inside the robot without straining each other or obstructing the joints is not an easy task. MERTZ's head design includes large cable passages through the center of head differential and the neck. This allows cable bundles to be neatly tucked inside the passages from the eyes all the way through to the base of the robot, thus minimizing cable displacement during joint movement. On the controller side, friction or locking connectors are used to ensure solid connections.

4.5. *Modular design*

The mechanical components of MERTZ were designed in a modular manner, or in a way that allows easy access to all components. This allows for easy maintenance and repair procedures without disassembly of large portions of the robot. One issue in maintenance is the wearing of components such as belts or drive cables. All the belts and drive cables on Mertz have been designed to take at least a 50 % greater load than they are required to, to reduce the possibility of breakage. If this does happen all belts and cables are replaceable with minor disassembly.

5. Preliminary Integrated System Architecture

Figure 4 illustrates the interconnections among the robot's hardware and software modules. The rectangular units represent the hardware components and the grey patches with rounded corners represent the software systems: sensorimotor, vision, and high-level behavior.

We paid careful attention to ensure that each layer of control is independent, such that the robot is safe-guarded upon removal of higher level control during run time. Motor control and behavior layers are implemented using embedded microprocessors, instead of more powerful but complex PCs, such that they can run autonomously and continuously at all times. With the absence of vision processing, the behavior system generates random motion commands to the robot. If the behavior system is also removed suddenly, the motor controller ensures that all joints stay

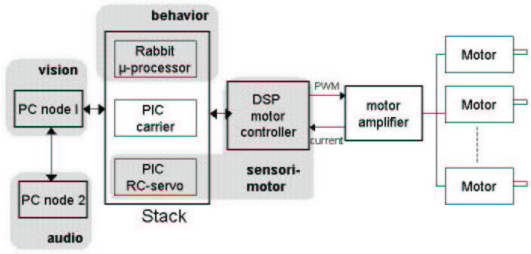

Fig. 4. The robot's preliminary integrated system architecture. Rectangular units represent hardware components. Grey patches with rounded corners represent software modules: vision, audio, behavior, and sensorimotor.

in place. In the event of power loss or cycle to the motor controller, the amplifier produces zero output to the motors until the motor controller is back into control. This control modularity comes very handy during the development and debugging process, because one can now be very sloppy about leaving the robot's motors on while updating and recompiling code.

5.1. *Hardware Systems*

Whereas our previous work has made use of off-the-shelf motion control products and PC nodes, in this project we implemented custom-made hardware for sensorimotor and behavior control. Off- the-shelf products, though powerful and convenient, are limited to a set of predetermined capabilities and can be unduly complex. Customizing our hardware to more precise specifications gives us greater control and flexibility. One caveat is that it may take some time to develop custom-made hardware to a reliable state.

5.1.1. *Motor Controller and Amplifier*

The motor controller is built using the Motorola DSP56F807. The controller supports PWM generation, encoder support, and A/D conversion for all existing axes. The amplifier uses the LMD18200 dual H-Bridge which accommodates up to 3A continuous output as well as current sensing. The ability to sense current is crucial as it provides a way to detect failures involving stalled motors. Particular attention was given to protect the robot against power cycle or shutdown. The motors and controller use separate power sources. Thus, we added simple circuitry to prevent the motors from running out of control if the controller happens to be off or reset, which could happen depending on the controller's initial state upon power reset.

5.1.2. *STACK*

The STACK [6] is a custom embedded architecture that was built by our group. It is a small footprint and expandable architecture consisting of up to 16 peripheral boards, interconnected through a bus to an 8-bit Rabbit 2000 microprocessor. The peripheral boards are designed to be physically stacked under the main Rabbit processor and communicate using a 9-bit RS-485 protocol through a common bus. [7] An 8 bit PIC microcontroller based carrier board is used to translate between the 9-bit RS-485 protocol to the main processor's 8-bit RS-232 protocol. We have implemented various addressable peripheral boards using PIC microcontrollers to provide a number of functions, such as interfacing to sensors and controlling actuators. In general, however, they can be built using any embedded processors as long as they can interface properly to the communication bus.

In Mertz, the stack consists of the DSP motor controller and a PIC-based board to interface with the RC servo motors. We are currently developing a new version of the STACK which will be communicating using the CAN bus protocol and utilizing an upgraded PIC microcontroller. We are also planning to upgrade the main processor to the much more powerful PowerPC controller.

5.1.3. *Vision and Audio Processing*

Mertz receives visual input from two color digital cameras with IEEE-1394 (FireWire) interface, chosen for their superior image quality. They produce 640 x 480 24 bit color images at the rate of 30 frames per second. Audio signal is perceived using a voice array desk microphone. A cluster of two Pentium based computers running Linux are used to interface with the camera and microphone. One of the computer nodes communicates with the STACK's main processor using the RS-232 protocol.

5.2. *Software Systems*

As mentioned above, all software systems are developed such that they can run continuously for many hours. Long term testing and experiments have been very helpful in identifying emergent and occasional bugs, as well as memory leaks.

5.2.1. *SensoryMotor Control*

Simple PD position and velocity control were implemented on the head and eye axes. Taking advantage of the Series Elastic Actuators, we use force feedback to implement force control for the neck joint. A simple PD position control was then placed on top of the force control. Various bounds are enforced to ensure that both position/force feedback and motor output stay within reasonable values. We plan to further augment the sensorimotor control to include more sophisticated fault detection and diagnosis.

Each axis is equipped with a potentiometer and a digital relative encoder. This allows for a fast and automatic calibration process, where upon startup, each axis is programmed to find its absolute position and then relies on the encoder for more precise position feedback. This streamlines the startup sequence to two steps which can be performed in any order: turning on the motor controller and turning on the motors. This is enough to get the robot to start moving around randomly. Higher level visual and audio modules can be independently started before or after turning on the robot. While the robot is running, motor controller can be reset at any time, causing the robot to re-calibrate to its default initial position and resume operation. The motors can also be turned off at any time, stopping the robot, and turned back on, letting the robot pick up where it left off.

5.2.2. *Vision and Audio*

MERTZ's vision system has to deal with a number of challenges in this project's first milestone. It has to be responsive to a large set of people who will be trying to interact with the robot. With minimal constraints and instructions, we expect to see a wide range of behaviors. In addition, it has to be exposed to different locations and thus varying lighting conditions. In order to encourage people to speak, the robot tries to mimic what people say to it. The voice array microphone is placed approximately 7 inches in front of the robot (see Figure 1) and is set up to accept input with a low enough energy threshold such that people can speak to the robot from up to 30 inches around the front half of the robot. As a result, background error and sometimes the robot's motor noise are unfortunately also included.

The vision system and communication among cluster PC nodes were implemented using YARP, an open source vision software platform developed by a collaboration effort between the MIT Humanoid Robotics Laboratory and LIRA-Lab at University of Genova. [14] YARP is a collection of vision code libraries, providing various image processing functionalities. YARP also supports message based interprocess communication distributed across multiple nodes. Processes can be dynamically started and connected to other existing processes. The audio system was implemented using the CMU Sphinx-2's phoneme recognition system [18] and the DECtalk phoneme-based speech synthesizer. [16] Each raw audio file and the corresponding phoneme sequence transcription are recorded during the experiment.

As an interface between MERTZ's vision system and its environment, we implemented a visual attention system to filter activities occuring in the field of view and compute the most salient target, similar to the system described in [4]. Inspired by Wolfe's model of attention [0], the system receives parallel input from low-level filters for saturated color, skin, face, and motion. Each filter's output is multiplied by individual weight and summed up to form a total saliency map. After the first phase of experiment, we quickly found that the attention system was easily confused and overwhelmed by the abundance of bright color and skin-like wooden interior in our new laboratory building.

We have so far implemented additional modules to allow the robot to be more responsive to people's interaction attempt and better in tracking salient targets. We use a Kanade Lucas Tomasi (KLT)-based tracker [21] to complement the frontal face detector developed by Paul Viola and Michael Jones. [0] Once a face is found, the robot tries to track it until either the face dissappears or the robot finds something else that is more interesting. The KLT-based tracker was obtained from Stan Birchfield's implementation that is publicly available [1]. Using the same approach to detect motion with active cameras [15], we also use the same KLT tracker to estimate displacement of background pixels due to robot motion at each frame. Motion is then calculated by subtracting current frame by the last frame translated by the estimated displacement. We also developed a color-histogram based tracker for tracking saturated-colored objects. When a patch of saturated color is detected, the system tries to segment a uniform color patch, calculate its color histogram, and track this color patch in subsequent frames. Since the background is always full of bright-colored walls in various colors, the tracker over time chooses to ignore certain patches that always remain static and thus probably belong to the background. A simple automatic camera gain setting procedure was implemented to ensure that all facial images share similar pixel contrasts under different lighting conditions.

5.2.3. *Behavior System*

The high level behavior system for Mertz is programmed in Creal (CREAture Language). [8] Creal is a programming language similar in concept to the earlier *Behavior Language* [9], L [10], and others [17]. These languages were specifically designed to implement behavior based programs in the incremental and concurrent spirit of the subsumption architecture. [11]

Creal allows users to create modules which are collections of threads sharing a local lexical environment. Threads are either scheduled to run periodically, when a condition is true, or on the occurence of some event combinations. Thread scheduler is not preemptive. Threads are scheduled round robin and each one is run whenever appropriate until it suspends. Our benchmark suggests that the Rabbit 2000 processors can support close to 1,000 threads, we have not tested using more than 50 threads on a single processor to date.

Each module is equipped with both input and output ports. Modules are interconnected by virtual wires linking output ports to input ports, carrying either 8 or 16 bit messages. As defined in the original subsumption architecture [11], wires may *suppress* and *inhibit* each other. Moreover, each peripheral board is referenced in Creal as a buffer which can be written to or read from.

As shown in Figure 5, the robot has a number of behavior modules. At the lowest level, module *random-explore* simply generates random motion commands to module *explore* which sends the commands to the robot's eyes. Module *vision-explore* receives input from vision, consisting of the x and y coordinates of the most salient area as determined by the attention system, relative changes in salient target's size

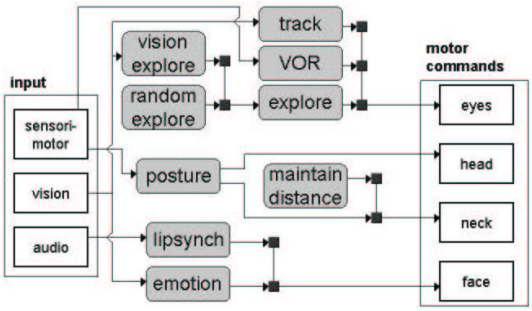

Fig. 5. MERTZ's behavior control for simple visual and verbal interaction with people. The robot orients to and track salient visual targets, receives audio input, and tries to mimic what people say. The behavior system was implemented in CREAL, a programming language specifically designed to implement behavior based programs in the incremental and concurrent spirit of the subsumption architecture.

and the type of current salient target. Whenever a salient target is present, module *vision-explore* forwards its input to module *explore*, inhibiting *random-explore* output. Module *VOR* monitors the eyes and head velocity and at times sends commands to the robot's eyes to compensate for the robot's moving head, inhibiting module *explore*. Module *track* receives tracking target information from vision whenever it is appropriate for the robot to track a face or color patch. Module *track* then outputs commands to the eyes, inhibiting module *VOR*. In parallel, module *posture* monitors the position of the robot's eyes and generates postural commands to the head and neck such that the robot is always facing the target. Module *maintain-distance* at times inhibits module *posture* to move the robot's neck to maintain a comfortable distance with target, estimated using the relative changes in salient target's size. Module *emotion* contains a very simple emotional model, where the robot's arousal, valence, and stance variables are either increased or decreased over time as a response to changes in visual and audio input. It then sends commands to the robot's eyebrows and lips to form facial expression. Lastly, module *lipsynch* receives phoneme sequences that the robot is currently uttering and commands the robot's lips to move accordingly.

6. Experiment

We conducted two phases of experiment. In the first phase, we equipped the robot with a simpler behavior system where it simply orients to salient visual targets, i.e. faces, skin color, and saturated colors. At the time, the robot's face was still in the design phase. The goal was to test the robot's robustness and study failure modes while the robot operated in its full range of motion. The robot ran for 47 hours within 4 days at different locations. We also collected raw visual data to observe

	Time	Duration	Location
Day 1	2pm – 10pm	8 hours	Laboratory
Day 2	12pm – 6pm	6 hours	Building Lobby
Day 3	10.30am – 11.30pm	13 hours	Balcony overlooking a student lounge
Day 4	9.30am – 4.30am	19 hours	Laboratory and moved to another area in the lab at 2 am

Fig. 6. Phase I experiment schedule, time, and location.

variations across different times and locations.

Based on lessons learned in the first phase, we further developed the robot to engage in simple visual and verbal interaction with people. In the second phase, the robot ran for 35 hours within 5 days at different public spaces, interacting with a large number of passersby.

6.1. *Phase I Setup*

The robot was setup to run at the laboratory and three other locations in the building. All of the head and neck axes, minus the eyelids and face plate, were actuated. Experiment schedule is shown in Table 6. The shortest and longest duration are 6 and 19 hours respectively. Initially, the experiment was conducted with supervision. As the robot showed a reasonable level of reliability (with the exception of the neck joint that has to be rested every couple hours), we started leaving the robot alone and checked on it every hour. During the experiment, people were allowed to approach but not touch the robot. While unsupervised, a sign was placed near the robot to prohibit people from touching it.

6.2. *Phase I Results*

The head and eye axes are so far free of failures. Most of the failures originated from the mechanical failures on the neck SEA actuators. Table 7 lists each failure that occurred during the experiment.

All observed failures involve the neck joint and its Series Elastic Actuators. Loose screws seem to be particularly problematic. A probable explanation is that the neck actuators are constantly in motion. The force control loop produces output that is proportional to the linear pot signal plus some noise. A series of filter have been placed in order to minimize noise. In addition, the load of each SEA motor is very small causing the control output to be very sensitive to even a trivial amount of noise. A dead-band was placed in software to reduce this effects, which eliminates

	Hours after startup	Failure
Day 1	2	The two motors actuating the neck's Series Elastic Actuators started heating up.
Day 2	1	One of the Series Elastic Actuators popped out of the neck joint because of a loose set screw.
Day 3	7	A wire connecting the linear potentiometer signal on the SEA to a signal conditioning board is loose.
	10	A screw was found missing in one of the SEAs, causing the motor to stall and heat up very quickly.
Day 4	0	At startup, we found that the potentiometer placed on the neck's differential tilt joint has been un-calibrated because of a loose screw. Each axis is relying on its potentiometer to calibrate itself to a default initial configuration upon startup.

Fig. 7. List of observed failures during experiment. All failures originate from the neck joint and its Series Elastic Actuators.

some but not all of the actuator's jitter. An alternative would be to put additional protection for the screws, i.e. using loctite on as many screws as possible. In addition, a better control mechanism for the SEA is definitely essential.

Collected visual data suggests that visual input varies greatly during the course of a day as well as at different locations (see Figure 8), which have different backgrounds, color schemes, and relative scale between objects. Moreover, the vision system seems to work best in the laboratory where it was developed. At the new experiment locations, unexpected features, such as a large bright orange wall, tend to overwhelm the attention system such that it is very difficult to attract the robot's attention.

6.3. *Phase II Setup*

At this time, the robot had been further developed to engage in simple visual and verbal interaction. The face plate with actuated eye brows and lips were added to the robot. The robot ran from 11 am to 6 pm for 5 days at different public spaces. A written sign was placed on the robot: " Hello, my name is Mertz. Please interact with me. I can see, hear, and will try to mimic what you say to me." The sign also explains that this is an experiment to test how well the robot operates in different environments and warns that the robot will be collecting face images and audio samples. A set of bright colored toys were placed around the robot. We monitored the robot from a distance to encourage people to freely interact with the robot, instead of approaching us for questions. Figure 9 shows the robot on day 5 of the experiment.

Fig. 8. On the top are examples of visual input taken from the same location at different times of the day. On the bottom are examples of visual input from different locations in the building.

6.4. *Phase II Results*

The robot ran without failures for the first 3 days of the experiment. On the 4th day, we detached one of the neck's SEA which seemed to be exposed to more friction than the other one, tightened the screws, and re-attached it. We also had to re-calibrate the motor control software to adapt to the resulting mechanical changes. The robot continued running with no problems for the rest of the experiment.

The robot was approached by an average of 140 people a day. The robot interacted with one invidual, small groups, and at times large groups of up to 20 people. People frequently spoke to the robot and tried to attract the robot's attention by moving their hands or the robot's toys around the robot. Each interaction session ranges from 30 seconds to 5 minutes long. While the robot interacted with people, it collected at least 100,000 tracked faces from at least 600 individuals and 6937 audio samples. Figure 10 shows some examples of the robot's attentional system's output during interaction with people.

Fig. 9. A sample interaction session on day 5 of the Phase II experiment. On the bottom right corner is a full-view of the robot's platform in the lobby of the MIT Stata Center.

7. Research Direction

Our long term research goal is to explore a more scalable learning framework, inspired by how human infants learn. In the first three years of life, the human brain produces a large number of synapses at an astonishing rate and human infants miraculously learn how to walk, manipulate objects, and acquire language. These advanced skills are constructed based on simpler layers of behavior, which the infant acquires and accumulates throughout the process. With the help of adults who regulate and provide scaffolding to the environment, infants can recognize their caregivers' voice at the age of two months, detect their own names as well as associate "mommy" and "daddy" to their parents at the age of six months, detect other familiar words soon afterward, and discover that a particular action like crying cause others to respond in certain ways. Common to these examples is the infant's basic ability to detect frequently occurring perceptual events and associate co-occurring external and self-initiated events.

In this project, we plan to explore how to implement this basic ability in a robot, assuming that human adult assistance is also available. The robot will be situated in *our* social settings, operating continuously for long periods of time in different public spaces, interacting with various people. We intend for Mertz to

Fig. 10. Examples of the attentional system's output while the robot interacts with multiple people on day 2 of the Phase II experiment. On the first and third columns are the robot's visual input. On the second and fourth columns are the corresponding output of the attention system indicating where the robot should be orienting to and tracking

incrementally learn to *remember* different individuals, associate frequently uttered phoneme sequences with corresponding visual targets, etc, through this direct and natural interaction. Aligned with Chapman's hypothesis that language acquisition cannot take place in the absence of shared social and situational contexts [13], we postulate that the act of experiencing the world may be a crucial step of the learning process. Long periods of continuous running time is particularly important in this context. Consequently, in order to *experience* the world, the robot must be robust enough to actually operate in human time.

8. Comparison to Related Work

HERMES is an autonomous humanoid service robot, designed specifically for dependability. It has been tested outside the laboratory environment for long periods of time, including an over six-month long exhibition in a museum where the robot is allowed to interact with the public. Although our project is exploring a different research direction, we fully concur with the underlying theme of increasing robot robustness and reliability.

Reliability is also a relevant topic in other museum tour-guide robots. [12,0,19] Deployment to a public venue and the need to operate on a daily basis naturally place reliability demands on these robots. Again, even though our robot is quite

different in form and function, we are exploiting a similar demand to have the robot perform on a daily basis and interact with many people in a public venue.

Kismet is an expressive active vision head robot, developed to engage people in natural and expressive face-to-face interaction. [3] The idea is that we can bootstrap from these social competence to allow people to provide scaffolding to teach the robot and facilitate learning. In order for this to happen, the robot would need to be up and running in the midst of people on a daily basis, observing and learning from what it perceives. This would also allow a more dynamic social interaction to occur. These are essentially the long term research goals of our project.

WE-4R is an emotion expressive humanoid robot, developed to explore new mechanisms and functions for natural communication between humanoid robot and humans [22]. In our project, the social interaction interface between the robot and people is also very crucial. We plan to explore this aspect in a more fundamental way once the robot is reliable and robust enough to be placed in humans' natural environment and operate continuously in human's time scale.

9. Conclusion

We have designed and built MERTZ, an active-vision humanoid head robot for exploring incremental online robot learning in a social context. We report on the design and development steps directed toward our first milestone, i.e. having the robot operate continuously for many hours in different public spaces and engage in simple visual and verbal interaction with a large number of people with minimal constraints. We have given particular attention to reliability and robustness issues during the design and construction of the robot. We plan to continue identifying failure modes throughout the rest of the robot development process in order to implement the appropriate fault detection and increase fault tolerance.

References

1. S. Birchfield, KLT: An implementation of the Kanade-Lucas-Tomasi feature tracker, http://www.ces.clemson.edu/ stb/klt/.
2. R. Bischoff, V. Graefe, Design Principles for Dependable Robotic Assistants, International Journal of Humanoid Robotics, vol. 1, no. 1 (2004) 95–125.
3. C. Breazeal, Sociable Machines: Expressive Social Exchange Between Humans and Robots, Sc.D. dissertation, Department of EECS, MIT, 2000.
4. C. Breazeal and B. Scassellati, A context-dependent attention system for a social robot. In Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI99). Stockholm, Sweden. 1146–1151, 1999.
5. R. Brooks, C. Breazeal, M. Marjanovic, B. Scassellati and M. Williamson, The Cog Project: Building a Humanoid Robot, in C. Nehaniv, ed., Computation for Metaphors, Analogy and Agents, Vol. 1562 of Springer Lecture Notes in Artificial Intelligence, Springer-Verlag, 1998.
6. R. A. Brooks, et al., ALIVE STACK User Manual, February 2003, unpublished.
7. R. A. Brooks, Proposed CREAL Communication Protocol - V1.0, July 2002.
8. R. A. Brooks, Creature language, Sep. 2003, http://www.ai.mit.edu/people/brooks/creal.pdf.

9. R. A. Brooks, The behavior language user's guide, AI Memo 1227, MIT Artificial Intelligence Lab, Cambridge, Massachusetts, Apr. 1990.
10. R. A. Brooks, C. Rosenberg, L -a common lisp for embedded systems, in Association of Lisp Users Meeting and Workshop, LUV'95 (1995) .
11. R. A. Brooks, A robust layered control system for a mobile robot, *IEEE Journal of Robotics and Automation* **2** (1986) (1) 14–23.
12. W. Burgard, D. Fox, D. Hhnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun and A.B. Cremers, Real Robots for the Real World – The RHINO Museum Tour-Guide Project, Proc. of the AAAI 1998 Spring Symposium on Integrating Robotics Research, Taking the Next Leap, Stanford, CA, 1998.
13. R.S. Chapman, Comprehension strategies in children, in Speech and Language in the Laboratory, School, and Clinic, J. Kavanaugh and W. Strange, eds. Cambridge, MA: MIT Press, 1978.
14. P. Fitzpatrick and G. Metta. http://sourceforge.net/projects/yarp0
15. G.L. Foresti and C. Micheloni, Real-time video-surveillance by an active camera, Ottavo Convegno Associazione Italiana Intelligenza Artificiale (AI*IA) - Workshop sulla Percezione e Visione nelle Macchine, Universita di Siena, September 11-13, 2002.
16. W. Hallahan, DECtalk Software: Text-to-Speech Technology and Implementation , http://research.compaq.com/wrl/DECarchives/DTJ/DTJK01.
17. I. Horswill, Functional programming of behavior-based systems, *Autonomous Robots* **9** (2000) (1) 83–93.
18. R. Mosur, Sphinx-II User Guide, http://cmusphinx.sourceforge.net/sphinx2.
19. I. Nourbakhsh, C. Kunz, T. Willeke, The Mobot Museum Robot Installations: A Five Year Experiment, In Proceedings of IROS 2003, Las Vegas.
20. J. E. Pratt, Virtual model control of a biped walking robot, Tech. Rep. AITR-1581, MIT Artificial Intelligence Laboratory, Cambridge, MA, USA, 1995.
21. J. Shi and C. Tomasi, Good features to track, IEEE Conference on Computer Vision and Pattern Recognition, pages 593-600, 1994.
22. H. Miwa, T. Okuchi, H. Takanobu, A. Takanishi, Development of a new human-like head robot WE-4", Proceedings of the 2002 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.2443-2448, 2002.
23. S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Halnel, C. Rosenbert, N. Roy, J. Schultze, and D. Schulz, MINERVA: A second-generation museum tour-guide robot, Proceedings 1999 IEEE International Conference on Robotics and Automation, vol 3, pages 1999-2005, May 1999.
24. A.M. Turing, Computing machinery and intelligence, Mind, 59, 433-460, 1950.
25. P. Viola, M. Jones, Robust Real-time Object Detection. Technical Report Series, CRL2001/01. Cambridge Research Laboratory, 2001.
26. J.M. Wolfe, Guided Search 2.0: A revised model of visual search. Psychonomic Bulletin and Review,1(2):202-238, 1994.